THE BELL SYSTEM TECHNICAL JOURNAL

DEVOTED TO THE SCIENTIFIC AND ENGINEERING ASPECTS OF ELECTRICAL COMMUNICATION

Volume 59 December 1980 Number 10

Improved Decoding Scheme for Frequency-Hopped Multilevel FSK System

By U. TIMOR

(Manuscript received December 19, 1979)

We have recently examined, for possible application to digital mobile radio telephony, a digital spread-spectrum technique employing multiple frequency-shift keying (MFSK) modulation with code-division-multiple access (CDMA) by frequency-hopping over a common bandwidth. The system uses the cellular approach where all mobiles within a cell communicate with a fixed base station in the cell. An analysis of base-to-mobile transmission shows that mutual interference limits the number of users which the system can accommodate at a given error rate. This paper describes a new decoding scheme to reduce mutual interference which makes use of the well-defined algebraic structure of the users' addresses. Analysis of the new decoder at high signal to noise (s/n) ratio shows it to outperform conventional decoding, allowing a 50 to 60 percent increase in the number of users who can simultaneously share the system at a given error rate. We describe a simple implementation of the decoder using shift registers.

I. INTRODUCTION

We have examined a digital spread-spectrum modulation technique employing frequency-hopping and multiple frequency-shift keying (MFSK) for multiple-access satellite communication[1] and for digital mobile radio telephony.[2] Every T seconds each user conveys a K-bit message by transmitting a sequence of L tones chosen from an alphabet of 2^K sine waves of duration $\tau (= T/L)$. Code-division multiplexing is

used, and the message is modulated onto the address (code) assigned to each user.

The receiver, knowing the address, can decode tbe received signal and extract the message. However, transmissions by other users can combine to cause an erroneous message, resulting in an ambiguous reception. Thus even without cbannel impairments, the number of simultaneous users the system can support at a given error probability is interference limited.

The interference can be minimized by a proper choice of addresses with minimum cross correlation. Schemes for assigning 2^K addresses which guarantee minimum mutual interference between 2^K or fewer users have been proposed.[3]

The performance of the system could be improved, in principle, by decoding the addresses of all possible users and analyzing the interference pattern. Such a complete decoding scheme is very complex.

This paper describes a simpler decoder which makes use of the well-defined algebraic structure of the sequences[3] to eliminate erroneous messages which come from interference. The result is a significant improvement in the performance of the system with a much smaller complexity than complete decoding. If, for example, the system has a total one-way bandwidth of 20 MHz and the data rate of each user is 32 kilobit/second, the number of users which the system can support at a bit error probability of 10^{-3} is increased by 60 percent over that attainable with conventional decoding.

In the performance analysis of the new decoder, an ideal (error-free) channel has been assumed. The conventional scheme has been further analyzed[2] for base-to-mobile transmission through noisy and multipath fading channels and was found to degrade gracefully. Under the same conditions, the new decoder is expected to have a similar degradation attaining its substantial improvements over the conventional decoder.

II. SYSTEM DESCRIPTION

The elementary signals of the system are a set of 2^K sine waves which are orthogonal over the chip duration τ. Every T seconds each user conveys a K-bit message by transmitting a sequence of L tones of duration τ chosen from the signal set. The sequence is uniquely determined by the user's address and the K-bit message and can be described as a pattern in the $L \times 2^K$ time-frequency matrix A. Simultaneous transmissions by M users can result in up to $L \times M$ entries in A (Fig. 1).

Let the address of the mth user be denoted by a vector $\mathbf{a}_m$:

$$\mathbf{a}_m = (a_{m1}, a_{m2}, \cdots, a_{mL}),$$

where each a_{mi} is a K-bit number corresponding to one of the 2^K frequencies of the system.

The transmitted sequence $\mathbf{Y}_m$ is formed by modulating the message X_m onto $\mathbf{a}_m$. The message can modulate the address in different ways. In Refs. 1 and 2, this is done by summing (mod 2^K) the address and the message to obtain

$$\mathbf{Y}_m = \mathbf{a}_m + X_m \cdot \mathbf{1}, \tag{1}$$

where

$$\mathbf{1} = \underbrace{(1, 1, \cdots, 1)}_{L}.$$

Other schemes are possible (see, for example, Ref. 3) as long as $\mathbf{Y}_m$ can be uniquely decoded.

Every τ seconds the receiver performs a spectral analysis of the composite received signal Y and decides which of the 2^K frequency cells contain energy. Thus after T seconds, assuming no channel impairments, a duplicate of A is generated at the receiver.

To decode X_m the receiver performs the inverse operation of (1), i.e., cycle-shift each column i of A by $-a_{mi}$ to obtain user m's decoded matrix A_m (Fig. 1). The message X_m will appear as a complete row in A_m.

Ambiguous decoding occurs when transmissions by other users combine to form other complete rows in A_m. In a previous analysis of the system,[1,2] it was assumed that if there is more than one complete row the receiver has no way of finding the correct message and has to

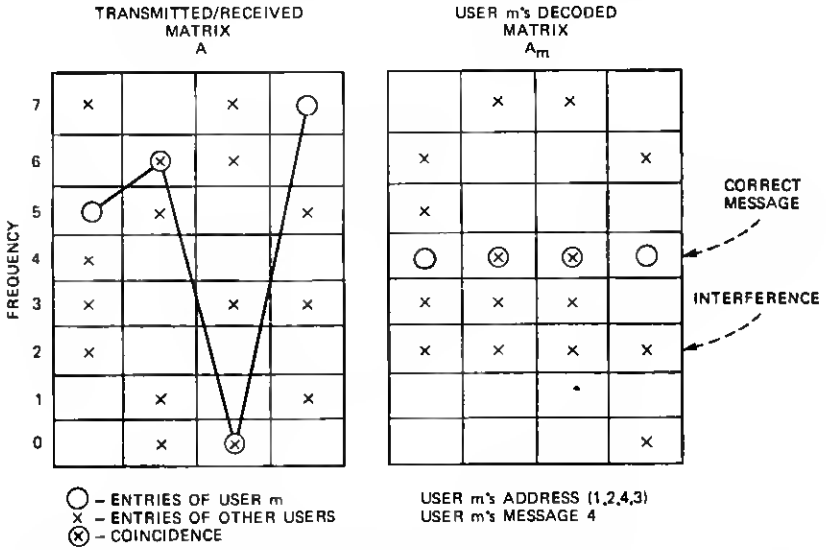


Fig. 1—Transmitted/received matrix A and user m's decoded matrix A_m for $K = 3$ and $L = 4$.

choose one of them at random. We show how further investigation of the complete rows can improve the performance.

III. ADDRESS ASSIGNMENT

It is convenient to describe the modulation/decoding procedure in algebraic terms by denoting the 2^K frequencies as elements of $GF(2^K)$, which is the finite field (Galois field) of 2^K elements $0, 1, \cdots, 2^K - 1$. Such finite fields exist for all $Q = p^n$, where p is a prime and $n = 1, 2, \cdots$, and the following analysis and results can be extended to systems with Q frequencies.

Accordingly, the K-bit message X_m, the components of $\mathbf{a}_m$, $\mathbf{Y}_m$, and the row numbers in A, A_m can all be expressed as elements of $GF(2^K)$, and the operations of addition and multiplication are performed according to the rules of $GF(2^K)$ (see the appendix).

Reference 3 describes an optimum structure of addresses which guarantees minimum mutual interference between 2^K or fewer users. In this scheme, each user is assigned a distinct element of $GF(2^K)$. Let γ_m denote the element assigned to user m and let β be a fixed primitive element in $GF(2^K)$. The address of user m is defined to be

$$\mathbf{a}_m = (\gamma_m, \gamma_m\beta, \gamma_m\beta^2, \cdots, \gamma_m\beta^{L-1}). \tag{2}$$

Since there are 2^K distinct elements γ_m, the maximum number of users (distinct addresses) is 2^K. Assuming word synchronization (i.e., all users start their sequences at the same time slot) it has been shown that for $L \leq 2^K - 1$ the transmitted sequences of any two users will coincide in at most one chip for all possible message values.

As mentioned before, other modulation schemes are possible (e.g., $\mathbf{Y}_m = \{y_{mi}\}, y_{mi} = X_m\beta^{i-1} + \gamma_m$), and the analysis and results that follow can be extended to such systems.

IV. NEW DECODING SCHEME

4.1 Decoding procedure

Let M be the number of simultaneous users of the system. Each user i transmits a sequence

$$\mathbf{Y}_i = \mathbf{a}_i + X_i \cdot \mathbf{1},$$

where the address $\mathbf{a}_i$ is assigned according to (2) and the message value is X_i. We assume a synchronous transmission, i.e., all users start their sequences at the same time slot.

In the decoded matrix A_m of user m, X_m appears as a complete row $X_m \cdot \mathbf{1}$. Suppose we have another complete row X' in A_m which is the result of interference. According to the address assignment, each chip in X' must come from a different user, so at least L users $i_1, \cdots, i_L$ have combined to cause this interference. The user that contributes

the interference in column n of X' is denoted by i_n. (If more than one user caused this interference, i_n can be any one of them.)

From eqs. (1) and (2), the jth entry ($j = 1, \cdots, L$) of user i_n in A_m will be at row $c_n(j)$ and column j where

$$\begin{aligned} C_n(j) &= \gamma_{i_n}\beta^{j-1} + X_{i_n} - \gamma_m\beta^{j-1}, \\ &= \delta_n\beta^{j-1} + X_{i_n}, \qquad j = 1, \cdots, L, \end{aligned} \tag{3}$$

γ_m is the address of the mth user and

$$\delta_n \triangleq \gamma_{i_n} - \gamma_m. \tag{4}$$

Note that $\delta_n \neq 0$ for all interferers i_n, $n = 1, \cdots, L$.

$C_n(n)$ is the row in which the entry of user i_n appears at the nth column. By definition, this row is X'. So we have the following relation:

$$C_n(n) = \delta_n\beta^{n-1} + X_{i_n} = X'. \tag{5}$$

Let $D^*_{X'}$ be the matrix obtained by subtracting the number X' from all rows of A_m (equivalently, shifting row X' in A_m to row zero). At $D^*_{X'}$ the entry of user i_n at column j ($j = 1, \cdots, L$) will be at row $d_n(j)$. From (3) to (5) we have

$$\begin{aligned} d_n(j) &= c_n(j) - X' \\ &= \delta_n(\beta^{j-1} - \beta^{n-1}), \qquad n, j = 1, \cdots, L. \end{aligned} \tag{6}$$

Thus by shifting row X' to row zero, the locations of the entries of the interferers $i_1, \cdots, i_L$ are independent of their message values $X_{i_1}, \cdots, X_{i_L}$.

So if row X' in A_m was formed by L interferers with addresses $\delta_n + \gamma_m$, their entries will appear in $D^*_{X'}$ at rows $d_n(j)$ and column j ($j, n = 1, \cdots, L$). We thus have the following theorem:

Theorem: A necessary condition for row X' in A_m to be caused by interference is the existence of L nonzero numbers δ_n in $GF(2^K)$ such that all entries $(d_n(j), j)(n, j = 1, \cdots, L)$ appear in $D^*_{X'}$.

Using (6) we can write for $n = 1, \cdots, L-1$:

$$\begin{aligned} d_n(j) &= \delta_n\beta^{n-1}(\beta^{j-n} - 1), \\ d_n(n+1) &= \delta_n\beta^{n-1}(\beta - 1). \end{aligned} \tag{7}$$

Thus we have the following relation:

$$d_n(n+1) = f_{j-n}d_n(j), \tag{8}$$

where

$$f_{j-n} \triangleq \frac{\beta - 1}{\beta^{j-n} - 1}, \qquad j = 1, \cdots, L, \qquad j \neq n, \tag{9}$$

and all the f_{j-n} are independent of the particular user and are fixed for the system. To check for tbe existence of user i_n ($n = 1, \cdots, L-1$), we multiply all entries (i.e., tbeir row numbers) in column j ($j = 1, \cdots, L$, $j \neq n, j \neq n+1$) of $D_{X'}^{*}$ by f_{j-n} and look for a "complete" row (complete except for the nth term). If we find such a row, then a possible interferer contributing to column n of X' exists. We see in Section 4.2 that there is a simple way to realize this procedure.

To check for user i_L, we modify this procedure and use the formula

$$d_L(L-1) = f_{j-L}^{*} d_L(j), \qquad j = 1, \cdots, L-2, \tag{10}$$

where

$$f_{j-L}^{*} \triangleq \frac{1}{\beta} f_{j-L}. \tag{11}$$

We then multiply all entries in column j ($j = 1, \cdots, L-2$) by f_{j-L}^{*} and check for a complete row.

If the condition is satisfied for all $n = 1, \cdots, L$, we assume that row X' in A_m is the result of interference.

Using this decoding scheme, all interference rows X' will be identified as such. The correct row X_m will usually fail to satisfy the condition for some n and thus will be identified and decoded as the correct message.

4.2 Realization

Multiplication of two elements in $GF(2^K)$ is most easily performed by expressing each term as a power of a primitive element β in the field (see the appendix).

Let $D_{X'}$ denote the matrix $D_{X'}^{*}$ without row zero. Each row q in $D_{X'}$ can be expressed by its exponent q', that is,

$$q = \beta^{q'}, \qquad q, q' = 1, \cdots, 2^K - 1. \tag{12}$$

Since β is primitive in $GF(2^K)$, the transformation $q \rightarrow q'$ is a row permutation $D_{X'} \rightarrow P_{X'}$.

Similarly, we can express the multiplication factors f_i, f_i^{*} as powers of β:

$$f_i = \beta^{k_i}, \qquad f_i^{*} = \beta^{k_i - 1}. \tag{13}$$

Multiplying an element at row q and column j in $D_{X'}$ by f_{j-n} is equivalent to cycle-shifting the corresponding element in $P_{X'}$ (which appears at row q' and column j) k_{j-n} positions.

So to find the existence of a user i_n contributing to column n of X', we cycle-shift all entries in column j of $P_{X'}$ (for $j = 1, \cdots, L$, $j \neq n$, $j \neq n+1$) k_{j-n} positions and look for a complete row.

Note that the rule of permutation $q \rightarrow q'$ as well as the shifting operator k_i, $i = \pm 1, \pm 2, \cdots, \pm (L-1)$ are fixed for the system and

are independent of the interference pattern. Thus they can be computed once and stored at the receiver.

In comparing the complexity of this decoding scheme to complete decoding, we have here $L + L(L-2)$ cycle-shifting operations for each complete row of A_m, while in complete decoding we perform $L(2^K - 1)$ cycle-shifting operations. A similar relation holds for the number of checking complete rows. So the complexity of the proposed scheme is of the order of L^2 compared to an order of $L2^K$ for the complete decoding. Since $L \approx K \ll 2^K$, the new decoding scheme has a much smaller complexity than complete decoding, yet it achieves a substantial improvement in performance.

V. DECODING ALGORITHM OF USER *m*

5.1 *Preliminary computations*

These computations are independent of the received matrix and are identical for all users so they can be performed once and stored at the receiver.

(*i*) Express all nonzero elements $q \in GF(2^K)$ by their exponents q', i.e., $q = \beta^{q'}$, where β is the primitive element in $GF(2^K)$ used to generate the addresses. This results in a table of $2^K - 1$ pairs (q, q').

(*ii*) Compute $2(L-1)$ terms k_i, $i = \pm 1, \pm 2, \cdots, \pm(L-1)$, where

$$\beta^{k_i} = \frac{\beta - 1}{\beta^i - 1},$$

and generate the cycle-shifting matrix $S = \{S_{i,j}\}$,

$$S_{i,j} = \begin{cases} 0 & i = 1, \cdots, L \quad j = i \\ k_{j-i} & i = 1, \cdots, L-1 \quad j = 1, \cdots, L \\ & \qquad\qquad\quad j \neq i \\ k_{j-L} - 1 & i = L \quad j = 1, \cdots, L-1. \end{cases} \tag{14}$$

5.2 *Computation performed for each code word (L chips)*

(*i*) From the received matrix A, compute the decoded matrix A_m by cycle-shifting the columns of A according to the address a_m. If A_m contains only one complete row at X_m, it is decoded as the message and the procedure terminates. If there are I complete rows ($I \geq 2$), $X^{(1)}, \cdots, X^{(I)}$ we check each row $X^{(i)}$ as follows.

(*ii*) Subtract (the number of row) $X^{(i)}$ from all rows in A_m and discard row zero to obtain matrix D_X.

(*iii*) Perform row permutation of $D_X \rightarrow P_X$ according to the table of $2^K - 1$ pairs (q, q').

(*iv*) For each n ($n = 1, \cdots, L$), cycle-shift all columns j in P_X ($j = 1, \cdots, L$, $j \neq n, n+1$) $S_{n,j}$ positions, according to the cycle-

shifting matrix S [eq. (14)] and look for a complete row (except the term in the nth column).

If for some n we don't find such a complete row, we decode

$$X_m = X^{(i)},$$

and the procedure terminates. Otherwise we assume $X^{(i)}$ to be the result of interference and repeat steps *(ii)* to *(iv)* for $X^{(i+1)}$.

(v) If all rows $X^{(i)}(i = 1, \cdots, I)$ pass the interference test (this could happen, for example, if an interference row coincides with the correct message), we cannot determine the correct message and have to pick one of the $X^{(i)}$ at random, as is always done in conventional decoding.

A flow diagram of the decoding scheme is outlined in Fig. 2.

The average time needed to decode X_m can be reduced by performing the interference test in a different order. Instead of checking the complete rows sequentially (as described above), we can check the first chip of all complete rows, then the second chip, and so on, until we find a chip that does not pass the interference test. It can be shown that with this scheme the average number of chips checked will be smaller and thus the average decoding time will be reduced.

VI. DECODER IMPLEMENTATION

The decoder can be implemented either in software or in hardware. A software implementation is appropriate for a microprocessor-controlled receiver and is done according to the flow diagram of Fig. 2.

Since most decoding operations require shifting [cycle-shifting or subtraction in $GF(2^K)$] of columns of various matrices, a simple way to implement the decoder in hardware is by using shift registers.

The decoder is composed of L identical units $U_1, \cdots, U_L$ representing the L columns of the matrices and a single L-stage shift register used to count the number of entries in a row and find the complete rows (Fig. 3a). A possible implementation of the basic unit U_j is depicted in Fig. 3b. The left shift register system R_{j1} contains $K + 2$ registers of length 1, 1, 1, 1, 4, 8, $\cdots$, 2^{K-1} (for a total of 2^K cells) and K identical double-pole double-throw switches. When all switches are in position 1, cycle-shifting can be performed. To subtract [according to the rules of $GF(2^K)$] a number 2^{i-1}, each cell in (binary representation) position $(\delta_K, \cdots, \delta_i, \cdots, \delta_1)$ has to be shifted to position $(\delta_K, \cdots, \bar{\delta}_i, \cdots, \delta_1)$. For $1 \le i \le K - 1$, this is accomplished by shifting the register 2^K steps while changing the position of the switch number i every 2^{i-1} steps (for $i = 1, 2$ we start at position 1 and for $3 \le i \le K - 1$ at position 2). For $i = k$, the register is cycle-shifted 2^{K-1} steps.

In general, to subtract a number $N = (n_K, \cdots, n_1)$, all the switches corresponding to the nonzero coefficients of its binary representation (except n_K) are operated while the register is shifted 2^K steps. (Switch

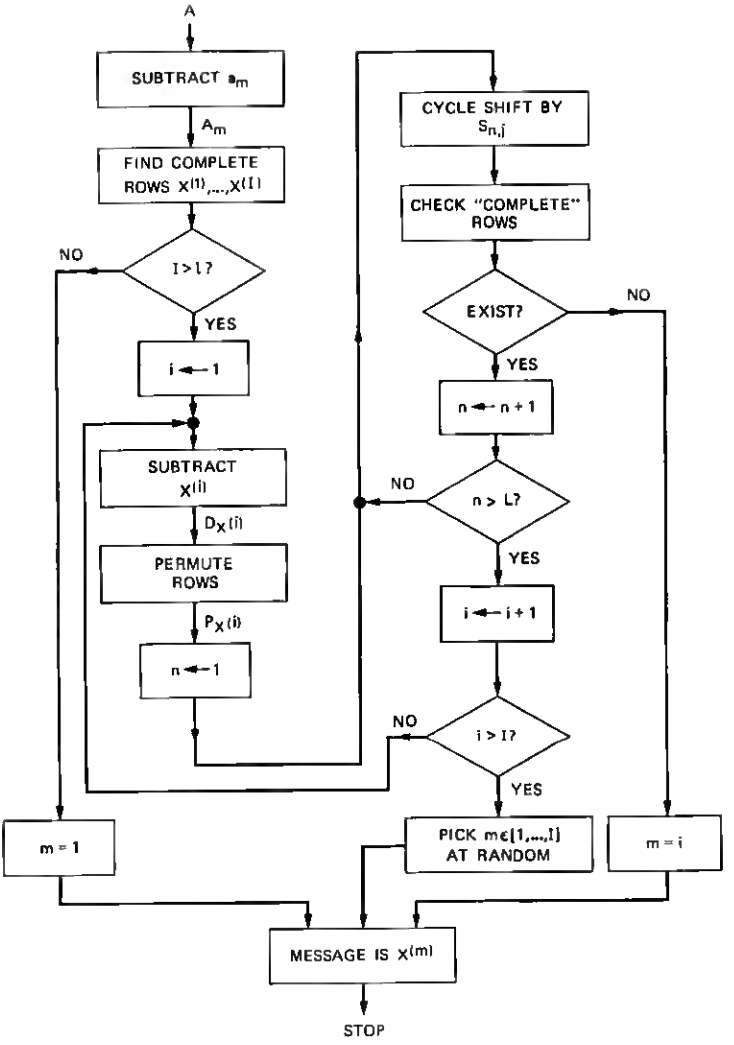
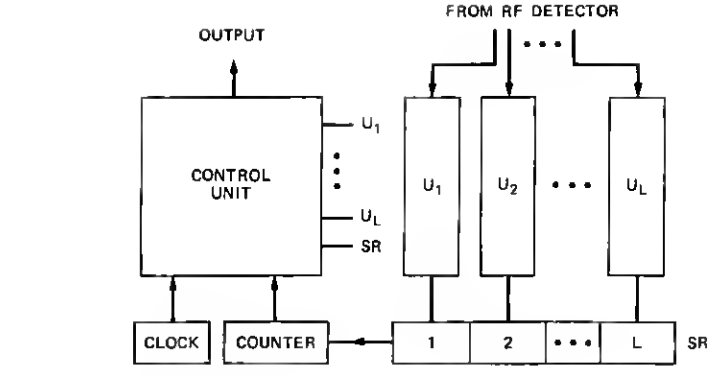


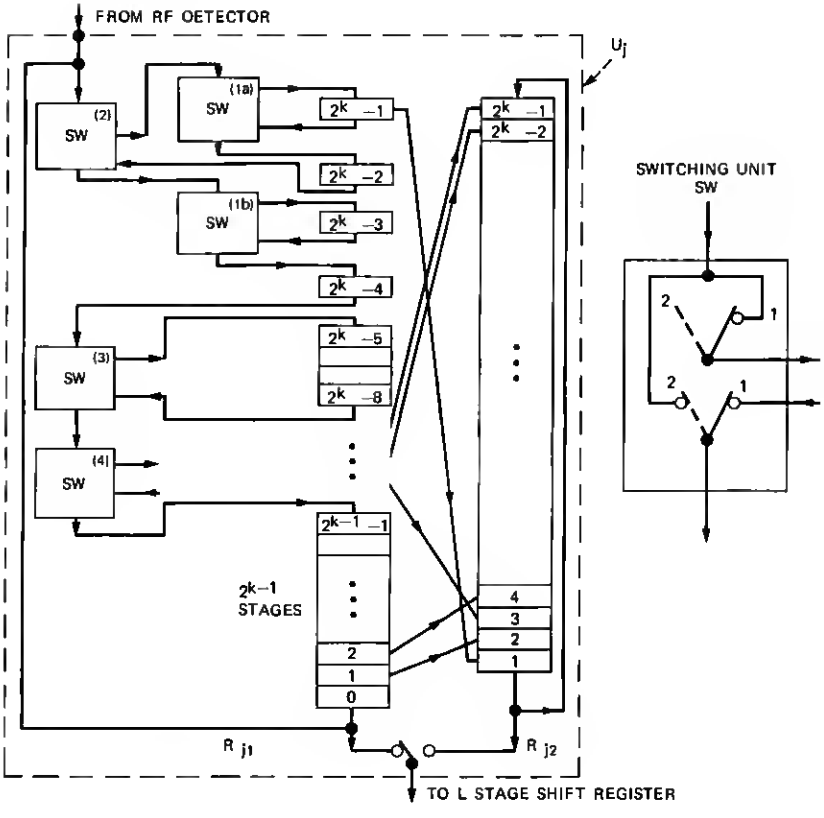
Fig. 2—Flow diagram of the new decoder. The input is the received matrix A. The output is a decoded message value X_m.

1b is used instead of 1a whenever switch 2 moves to position 2.) If $n_K = 1$, the register is cycle-shifted additional 2^{K-1} steps. An example for $N = 3$ and $N = 7$ is shown in Table I.

The right shift register R_{j2} contains $2^K - 1$ cells and performs cycle shift operation. The data are fed from the left register system to the right register in a parallel-to-parallel mode. Each stage in R_{j2} is hard-wired to a stage in R_{j1} according to the row permutation (fixed) rule $q \rightarrow q'$.



(a)



(b)

Fig. 3—Hardware implementation of the decoder. (a) L identical units U_j operate in parallel. SR is used to find the complete rows. (b) The basic unit U_j includes two shift register systems: R_{j1} can perform subtraction in $GF(2^K)$ using the switching units SW, and R_{j2} perform additional decoding, when necessary, to remove ambiguity.

Table I—$K = 3$ (8 cells, 3 switches)

	0	1	2	3	4	5	6	7	8	9	10	11	12
Switch													
1a		1	2	1	1	1	2	1	1	1	1	1	1
1b		1	1	1	2	1	1	1	2	1	1	1	1
2		1	1	2	2	1	1	2	2	1	1	1	1
Content of Register													
7	7	0	0	0	0	4	4	4	4	3	2	1	0
6	6	7	1	1	1	0	5	5	5	4	3	2	1
5	5	6	7	2	2	1	0	6	6	5	4	3	2
4	4	5	6	7	3	2	1	0	7	6	5	4	3
3	3	4	5	6	7	3	2	1	0	7	6	5	4
2	2	3	4	5	6	7	3	2	1	0	7	6	5
1	1	2	3	4	5	6	7	3	2	1	0	7	6
0	0	1	2	3	4	5	6	7	3	2	1	0	7
										Cycle shifting			
Subtract 3 = (1, 1)													
Subtract 7 = (1, 1, 1)													

To decode a block, the data are serially fed from the RF detectors to the left register systems R_{j1} ($j = 1, \cdots, L$), which are then shifted (with switch positions corresponding to the address a_{mj}) to generate A_m and find its complete rows. To check a complete row, it is shifted to row zero and the data are transferred to the right registers R_{j2}, which are then cycle-shifted according to the shifting matrix S.

The L-stage shift register SR is parallel-fed from either R_{j1} or R_{j2} and is used to count the number of entries in a row to find the complete rows.

VII. ERROR PROBABILITIES

Let the system parameters be K (2^K frequencies), L (sequence length), and M (simultaneous users). We assume an algebraic address assignment [such as (2)] and a synchronous transmission without channel impairments.

If the decoded matrix A_m of user m has only one complete row, this row is correctly decoded as the message value X_m. If there are two or more complete rows in A_m, an interference test is performed. An interference row will always pass the test since each of its chips is part of a sequence transmitted by some user. The correct row will usually fail to pass the test for some of its chips and thus can be identified and correctly decoded.

We will have an ambiguity in decoding when the following two conditions are satisfied:

(*i*) There are two or more complete rows in A_m.

(*ii*) The correct row passes the interference test.

When this happens, we choose one complete row at random and decode it as the message.

Let P_1 and P_2 be the probabilities of condition (i) and (ii), respectively. Although not strictly independent, it can be shown using random-coding arguments that the two conditions can be assumed to be independent with a negligible effect on the probability of error. P_1 can be upper bounded[1] by the union bound

$$P_1 < (2^K - 1)p^L, \tag{15}$$

where

$$p = 1 - (1 - 2^{-K})^{M-1}. \tag{16}$$

Condition (ii) can happen as follows: j chips in X_m ($j = 0, 1, \ldots, L$) are also the result of interference (by at least j other users) and the remaining $L - j$ chips, although not caused by interference, pass the interference test. That is, entries from other users combine to form complete rows in the cycle-shifted version of P_{X_m}.

Let $P_{2,j}$ be the probability of condition (ii) when exactly j chips in X_m come from interference, then

$$P_2 = \sum_{j=0}^{L} P_{2,j}. \tag{17}$$

$P_{2,L}$ corresponds to the case where interference from other users combine to form a complete row which coincides with X_m. Thus

$$P_{2,L} < p^L. \tag{18}$$

Similarly,

$$P_{2,L-1} < Lp^{L-1}(1 - p)\hat{p}, \tag{19}$$

where $Lp^{L-1}(1 - p)$ is the probability that exactly L -- 1 chips come from interference and $\hat{p}$ is the probability that a complete row ($L - 1$ entries) exists in the cycle-shifted P_{X_m}, when checked for the remaining chip. Since this complete row could be any of the $2^K - 1$ rows of P_{X_m}, we have

$$\hat{p} < (2^K - 1)p^{L-1}. \tag{20}$$

The general term $P_{2,L-j}$ can be upper bounded by

$$P_{2,L-j} < p^{L-j}(1 - p)^j \binom{L}{i} \hat{p}^j. \tag{21}$$

Thus

$$P_2 < p^L \sum_{j=0}^{L} \binom{L}{j} [(2^K - 1)(1 - p)p^{(L-2)}]^j,$$

$$P_2 < p^L(1 + S)^L, \tag{22}$$

where

$$S = (2^K - 1)(1 - p)p^{(L-2)}. \tag{23}$$

So, as long as $p(1 + S) < 1$, we can upper-bound the word error probability P_E by

$$P_E < \tfrac{1}{2} P_1 \cdot P_2 < \tfrac{1}{2} (2^K - 1)(1 + S)^L p^{2L}, \tag{24}$$

where the factor of ½ comes from the random choice of message when the decoding fails. From this we can obtain a bound on the bit error probability[1]

$$p_b < 2^{K-2}(1 + S)^L p^{2L}. \tag{25}$$

Comparing the performances to conventional decoding, we see that the new decoding scheme has nearly doubled the exponent of p.

VIII. SYSTEM PERFORMANCE

The bandwidth necessary to support 2^K tones that are orthogonal over $\tau = T/L$ seconds is

$$W = 2^K/\tau = 2^K L/T \text{ Hz}.$$

The transmission rate per user is

$$R = K/T \text{ bit/s}.$$

For a given W, R, and K, the length of the sequence L is determined by

$$L = rK2^{-K},$$

where

$$r = W/R.$$

Let M be the number of users that the system can accommodate at a given error probability P_b. The efficiency η of the system is defined as

$$\eta = MR/W = (M/r) \text{ bit/s/Hz},$$

the information rate (bit/s) per unit bandwidth transmitted through the system with a bit error probability P_b. For a given W, R, and P_b, there is an optimum K which maximizes the efficiency.

Figure 4 shows the dependence of M (and η) on K for $P_b = 10^{-3}$ and $r = 626, 320, 165$ (corresponding to $R = 32$ kilobit/second and $W \approx 20$, 10, and 5 MHz, respectively) using the upper bound (25) on P_b.

Since the decoding scheme is based on the address assignment (3) which can accommodate at most $2^K - 1$ users, M cannot exceed this number. This is shown by the dashed curves. As can be seen, this has no influence on the choice of optimum K and the performance at that K.

Figure 5 shows the efficiency of the system as a function of r for $300 \leq r \leq 700$, when the optimum K was taken at each r. It can be seen that, since K, L, and M assume only integer values, the curve is not

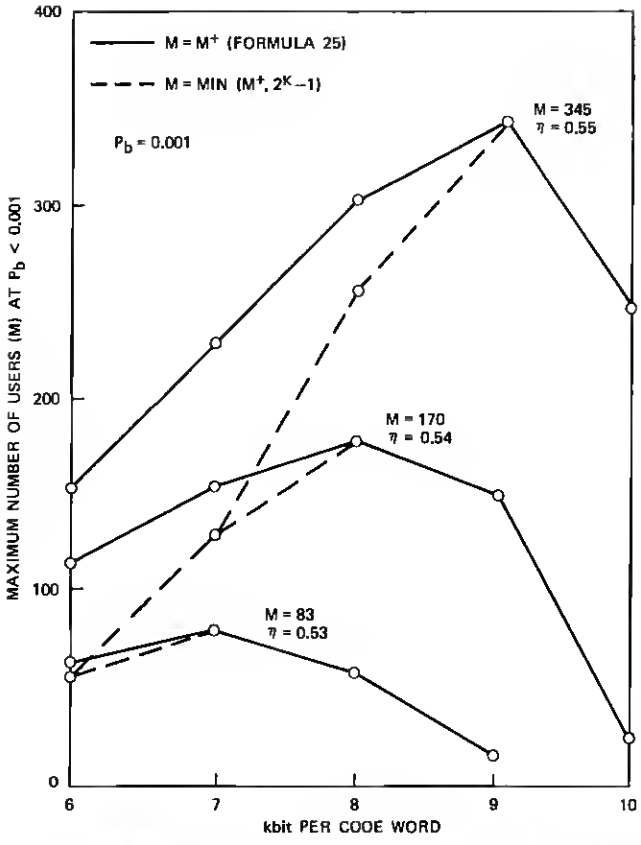


Fig. 4—Number of users M which can simultaneously share the system at an error probability of 0.001 as a function of K, for different values of $r = W/R$. The solid line (M^+) is computed using the upper bound (25). The broken line indicates the limit on M (2^K) when the address assignment (2) is used.

continuous but has local maxima at specific values of r. The same phenomenon occurs for conventional decoding.

To compare the performance of the new decoding scheme with that of conventional decoding, consider the case where $W = 20.025$ MHz, $R = 32$ kilobit/second ($r = 625.8$), and $P_b = 10^{-3}$. (At this r, the efficiencies of both schemes have local maxima.) The optimum K is 9 (corresponding to 512 frequencies) and $L = 11$ in both schemes. The maximum number of users which the system can accommodate is 345 in the new scheme compared to 216 with conventional decoding. The corresponding efficiencies are 55 and 34.5 percent, an improvement of almost 60 percent. A similar improvement (of 50 to 60 percent) is obtained throughout the range of r.

Figure 6 depicts the bit error probability as a function of the number

of users for both decoding schemes for the above parameters ($K = 9$, $L = 11$, and $r = 625.8$).

While the number of users which the system can actually support would be less than illustrated here because of the idealized model used in the analysis (as discussed in Ref. 2), the comparative results are believed to be valid.

XI. SUMMARY AND CONCLUSIONS

A new decoding scheme which makes use of tbe algebraic structure of the addresses to eliminate erroneous messages which come from interference was described. Analysis of tbe noiseless case, where interference from other users is the only source of performance impairments, shows a 50 to 60 percent improvement in efficiency over conventional decoding.

In noisy and multipath fading channels, the received matrix is corrupted by insertions and deletions of entries and the correct row might be incomplete. In this case, the row with the maximum number of entries is decoded as the correct message, and the probability of ambiguous reception is increased. For base-to-mobile transmission, the conventional decoder has been analyzed[2] in Gaussian noise and a multipath fading environment and found to degrade gracefully. Other

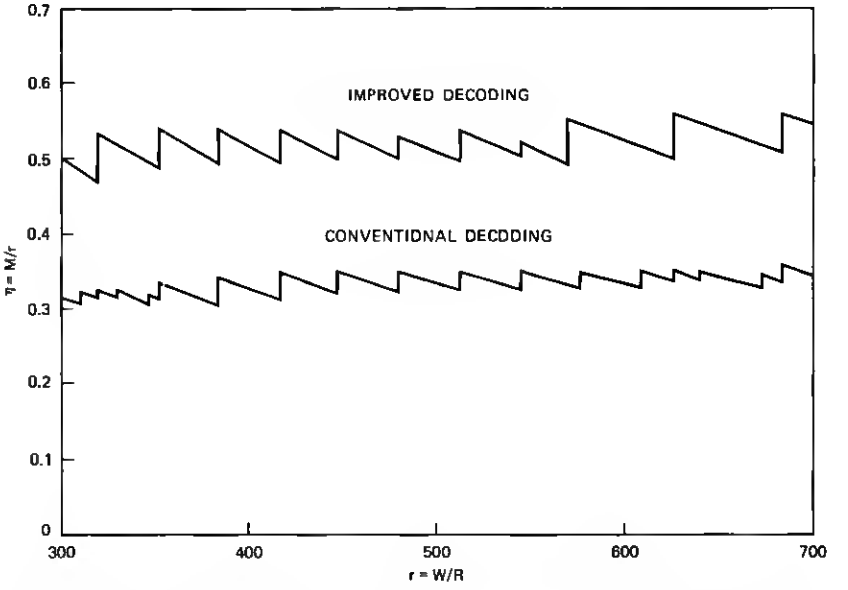


Fig. 5—The efficiencies ($\eta = M/r$) for both decoding schemes as a function of $r = W/R$, when the optimum K is taken at each r. Since K, L, and M assume only integer values the curve is not continuous.

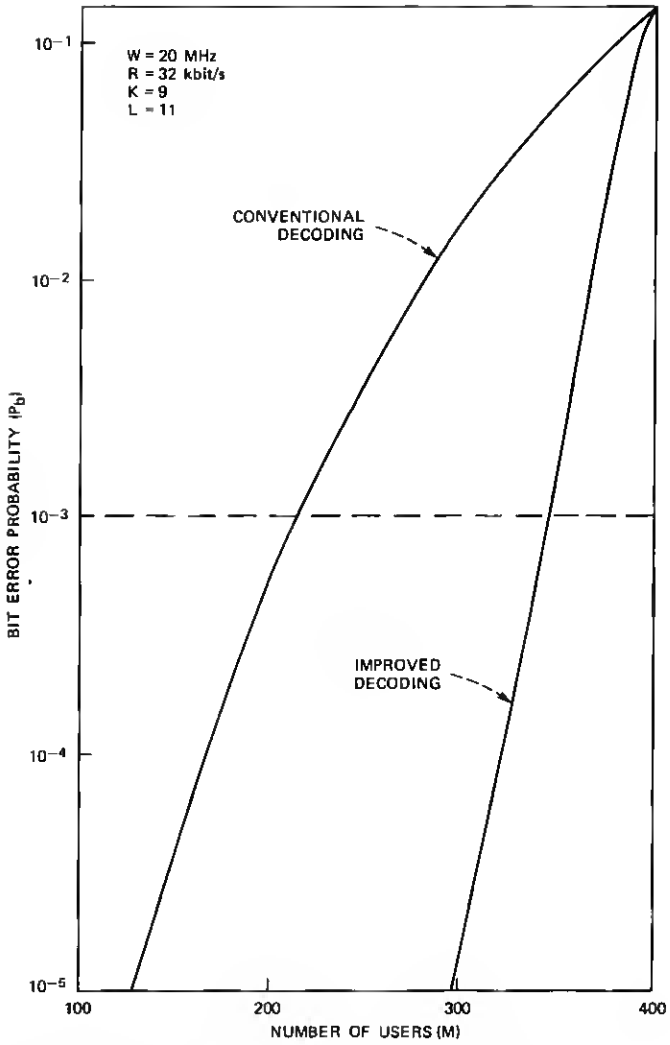


Fig. 6—Performance curve as a function of the number of users M for conventional and improved decoding. The system parameters are $W = 20$ MHz, $R = 32$ kilobit/second, $K = 9$, and $L = 11$.

factors such as shadow fading, intercell interference (when the system is embedded in a matrix of cells covering a large geographic area), and synchronization error due to multipath propagation delay spread (most relevant to mobile-to-base transmission) are likely to further reduce the number of users but as yet have not been analyzed. The new decoder is likely to have a similar degradation in typical noisy conditions encountered in mobile radio and satellite communication sys-

tems, but can be expected to yield a substantial improvement over conventional decoding.

APPENDIX

Algebra of Finite Fields

A.1 Rules of addition (subtraction) and multiplication (division)

Let $GF(Q)$ be a finite field (Galois field) of Q elements $0, 1, \cdots, (Q-1)$. Such fields exist for all $Q = p^n$, where p is a prime and $n = 1, 2, \cdots$.

If Q is a prime ($n = 1$), the rules of addition and multiplication are defined by ordinary modulo Q arithmetics.

If $Q = p^n$, $n = 2, 3, \cdots$ we first represent the elements as p-ary vectors of length n (if $p = 2$, this is the binary representation). Addition is now defined as mod p addition of the components. To specify multiplication, the n-tuplets are transformed into polynomials of degree $n-1$ in Z by letting the first digit be the coefficient of Z^{n-1}, and so on. The multiplication rule is polynomial multiplication (ordinary multiplication with mod p addition of coefficients) modulo an irreducible polynomial of degree n.

A.2 Multiplication using an exponent representation

A simple way to perform multiplication in $GF(p^n)$ is: Every nonzero element in the field can be expressed as a power of a primitive element β (β is primitive if the smallest integer k such that $\beta^k = 1$ is $k = Q-1$). Thus if we have a list (table) of all exponent representations, we can perform a multiplication of two numbers by adding, modulo $(Q-1)$, their exponents. This is analogous to multiplication of real numbers using a table of logarithms.

REFERENCES

1. A. J. Viterbi, "A Processing Satellite Transponder for Multiple Access by Low-Rate Mobile Users," Digital Satellite Communications Conference, Montreal, October 23–25, 1978.
2. D. J. Goodman, P. S. Henry, and V. K. Prabhu, "Frequency-Hopped Multilevel FSK for Mobile Radio," B.S.T.J., *59*, No. 7 (September 1980), pp. 1257–75.
3. G. Einarsson, "Address Assignment for a Time-Frequency-Coded, Spread-Spectrum System," B.S.T.J., *59*, No. 7 (September 1980), pp. 1241–55.